UNIVERSITÉ DE MONTRÉAL

CENTRE DE RECHERCHE SUR LES TRANSPORTS

CASE POSTALE 6128, SUCCURSALE A, MONTRÉAL, QUÉBEC, H3C 3J7

CRT-567

AN ALGORITHM FOR THE DESIGN OF MAILBOX COLLECTION ROUTES IN URBAN AREAS*

by

Gilbert Laporte[1]
Suzanne Chapleau[1]
Philippe-Eric Landry[2]
and
Hélène Mercure[3]

[1] Centre de recherche sur les transports, Université de Montréal, C.P. 6128, Succursale A, Montréal H3C 3J7, Canada.

[2] Canada Post Corporation, 545 rue MacArthur, Ville Saint-Laurent H4T 1T6, Canada.

[3] École des Hautes Études Commerciales de Montréal, 5255 avenue Decelles, Montréal H3T 1V6, Canada.

* The authors are grateful to the Canadian Natural Sciences and Engineering Research Council (grant A4747) and to the Canada Post Corporation for their support. Thanks are also due to the anonymous referees for their valuable comments.

Cette étude a été publiée grâce à une subvention du fonds F.C.A.R. pour l'aide et le soutien à la recherche.

Université de Montréal
Centre de recherche sur les transports - Publication #567
April 1988

Dépôt légal - 2e trimestre 1988

ABSTRACT

Mailbox collection is a complex and costly operation in most postal services. The authors recently undertook a study aimed at improving this operation at the Canada Post Corporation (CPC). This paper describes an improvement to the clustering and routing algorithm currently in use. Various traffic and union rules affecting mailbox collection in an urban context are first described and their effect on shortest distance computations is explained. This is followed by a description of the clustering and routing algorithm currently used at CPC. A model and a generalized travelling salesman algorithm which take the various rules into account are then presented. It is then shown how the CPC routes can sometimes be improved by using the proposed algorithm as a post-optimizer.

Key words : mailbox collection, vehicle routing, travelling salesman problem.

RÉSUMÉ

La levée des boîtes aux lettres constitue une opération complexe et coûteuse dans la plupart des services postaux. Les auteurs ont entrepris une étude visant à améliorer cette opération à la Société canadienne des postes (SCP). Cet article décrit une amélioration à l'algorithme de regroupement et de construction de tournées présentement en usage. On décrit d'abord certaines règles (de circulation et syndicales) affectant le calcul du chemin le plus court entre deux boîtes aux lettres. Vient ensuite une description de l'algorithme présentement utilisé par SCP. On présente ensuite un modèle et un algorithme du voyageur de commerce généralisé pour le problème. Finalement, on démontre comment cet algorithme utilisé comme post-optimiseur peut parfois donner lieu à une diminution des longueurs de tournées.

Mots-clés : levée des boîtes aux lettres, problèmes de tournées, problème du voyageur de commerce.

1. INTRODUCTION

Over the last decade, Canada Post Corporation (CPC) has initiated a number of projects aimed at computerizing its mail collection and delivery operations. Within this context, the authors recently undertook a study to improve mailbox collection operations in urban areas. Tests conducted on ten problems indicate that travel distance reductions can be achieved; in the test problems, these ranged from 0% to 14.5% and averaged 5.5%. Given the large costs involved in mailbox collection, this could translate into significant dollar savings. The proposed routing algorithm is now part of a broader implementation project contracted out by CPC to members of our group.

Routing problems arising from postal service operations can be classified into two broad categories: i) **arc routing** problems (e.g. door-to-door delivery — for recent results, see Levy (1987), Bodin and Levy (1987), Levy and Bodin (1988a, 1988b) and Roy and Rousseau (1988); ii) **node routing** problems. For surveys of node routing methodology, see Bodin et al. (1983) and Laporte and Nobert (1987).

Our problem belongs to the latter category. Since it concerns mailbox collection in urban contexts, several traffic rules such as left-turn and U-turn prohibitions have to be taken into account when designing the routes. The problem is further complicated by a union "no crossing rule": for safety reasons, drivers are not allowed to cross a street on foot in order to empty a mailbox. In the next section, we show how these prohibitions and rules affect postal truck route design in an urban context. Section 3 describes the current clustering and routing method used by CPC. In sections 4 and 5 we present a model and an algorithm which take these rules into account. In section 6, we show how the routes developed by CPC can sometimes be improved by using the proposed algorithm as a post-optimizer. The conclusion follows.

2. MAILBOX COLLECTION IN AN URBAN CONTEXT

Designing optimal mailbox collection routes usually involves two main steps:

(i) **clustering:** determine which mailboxes will belong to each vehicle route;

(ii) **routing:** generate a minimum length vehicle route through the mailboxes of each cluster.

Cluster and route design varies according to whether it is carried out in rural or in urban areas. Rural road configurations are far less complex than urban street networks and allow fewer routing alternatives. Moreover, they frequently have different traffic rules than cities (for example, left-turns are more often permitted on country roads). Our study was undertaken for urban contexts. In order to give an idea of the overall problem size, in a city like Montreal (approximate population: 3 million), there are 3,084 street mailboxes which must be emptied from 71 postal stations (depots) between 1 and 4 times a day. The number of vehicle routes starting and ending at each of these stations is most often equal to 1, 2 or 3.

Since not all mailboxes have the same collection frequency, they may be part of different clusters and routes at different times of the day. Clusters are defined for each collection period in such a way that all their mailboxes can be feasibly visited within a preset time period (60, 90 or 120 minutes). Typically, each cluster will include between 20 and 50 mailboxes. One important consideration in designing clusters is that they should be relatively compact, perhaps not in a visual sense, but with respect to a given distance function. In practice, most clusters tend to be visually compact and adjacent to a postal station.

This paper is concerned with the routing step. Route design is obviously affected by mailbox location. For example, locating several mailboxes on the same side of a street may help, at least partially, to deal with the no crossing rule. Carefully choosing mailbox locations so as to avoid unnecessary turns and detours may also yield overall travel distance reductions. However, mailbox relocations are not always possible nor advisable. An obvious reason is that since a given mailbox may belong to several vehicle routes within the same day, a location which is good for one route may not be adequate for another route. Moreover, mailbox relocations (even across the street) often meet with opposition from the various businesses and citizens' groups concerned so that, for the time being, CPC prefers to leave this issue untouched.

Another difficulty related to our routing problem is the definition of a distance matrix. While this task is relatively straightforward in most routing applications, it is made more difficult here by the various traffic and union rules. In particular, the shortest distance between two given mailboxes is no longer uniquely defined, but depends rather on the itinerary followed by the postal truck before arriving at the first mailbox.

Consider, for example, the street plan shown in figure 1. Assume there are no one-way streets. There are two mailboxes located at sites A and B. Mailbox A can be served either by a truck arriving from the South, or by a truck arriving from the West. In the first case, the shortest distance $c'_{A,B}$ from A to B is simply the length of the vertical arrow from A to B. However, in the second case, the truck will have to go around the block (as in figure 1a or 1b) before reaching B. Then, the shortest distance from A to B is $c''_{A,B}$. If U-turns were allowed, $c''_{A,B}$ would be computed as shown in figure 1c, and if crossings and left-turns were permitted, $c''_{A,B}$

would then be equal to $c'_{A,B}$ (figure 1d). In order to compute shortest distances, it is necessary in the model to represent all mailboxes located at intersections by two different stops. Thus, mailbox A would be replaced by A1, located on arc AD and by A2, located on arc CA. This yields a partition of all stops into three sets:

I all stops corresponding to mailboxes not located at street corners

J_1 all stops representing first copies of mailboxes originally located at intersections

J_2 all stops representing second copies of mailboxes originally located at intersections.

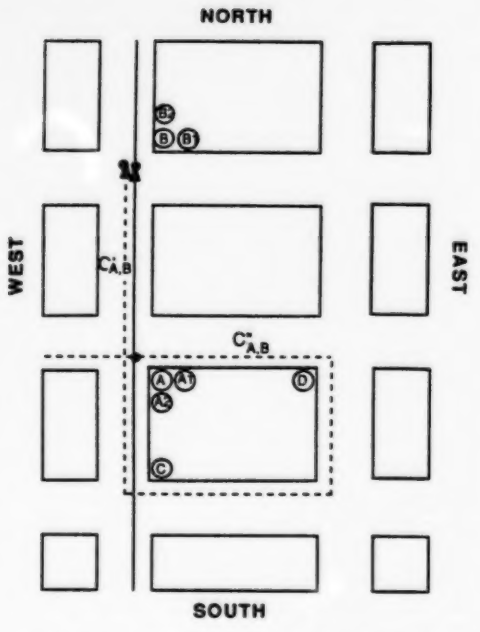
The shortest distance matrix between these stops can now be computed without ambiguity. In the example, $c_{A2,B2} = c'_{A,B}$; $c_{A2,B1} = c_{A1,B1} = c_{A1,B2} = c''_{A,B}$.

3. THE CLUSTERING AND ROUTING ALGORITHM USED BY CPC

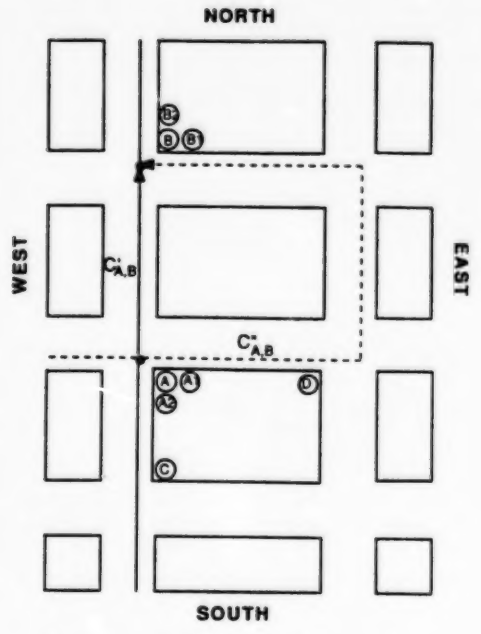
We summarize in this section the clustering and routing procedure in use at CPC.

Step 1 **Definition of a distance matrix.** Duplicate mailboxes located at street intersections and define the distance matrix on $(I \cup J_1 \cup J_2)^2$, as explained in section 2.

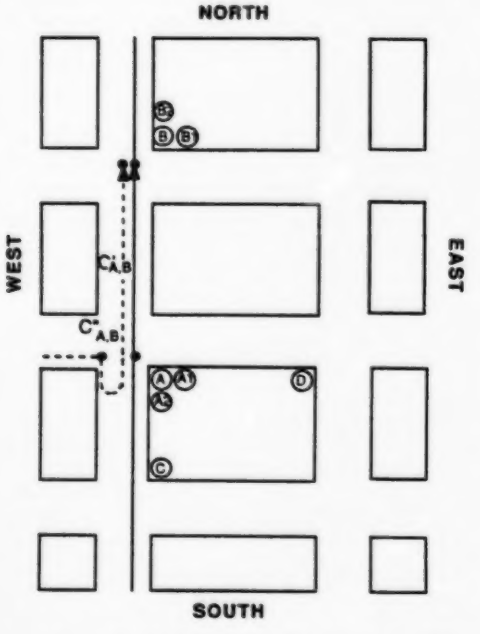
Step 2 **Initial clustering.** This step consists of defining clusters corresponding to all stops included on a single vehicle route. Cluster C_p will include n_p stops. Clusters are gradually built around the mailbox which is furthest from the post office and which is not already included in a cluster. A cluster is completed when



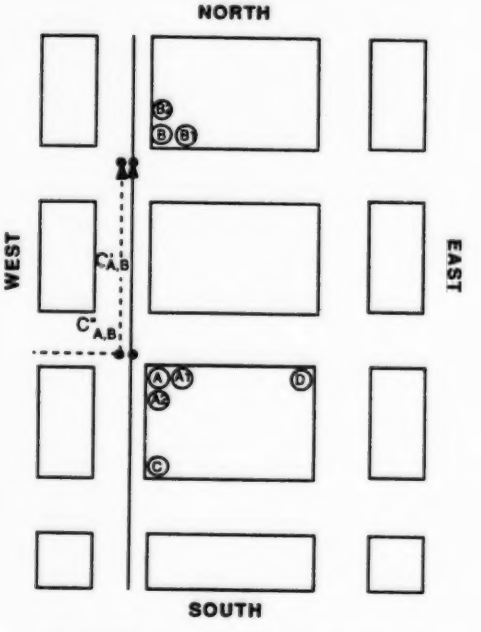
a) no U-turns, no left-turns, no crossings



b) no U-turns, left-turns allowed, no crossings



c) U-turns and left-turns allowed, no crossings



d) left-turns and crossings allowed

Figure 1: Shortest distance from A to B: street plan showing duplicated mailboxes

it contains enough mailboxes or when the estimated time required to service these mailboxes is reached. At the end of this step, all clusters have been defined, but they may not be compact.

Step 3 **Updated clustering.** Determine the centroid of each cluster, i.e. the mailbox for which the sum f_p of squared distances to all remaining mailboxes in the cluster is minimized. As in Bourjolly et al. (1981), perform marginal modifications to the clusters (and while doing so, redefine the centroids) as long as $\sum_p f_p$ can be reduced. These modifications are of two kinds: (i) simple moves of stops from one cluster to an adjacent one; (ii) exchanges between two adjacent clusters. At the end of this step, we have obtained compact clusters with respect to the distance matrix defined in step 1.

Step 4 **Route generation.** Within each cluster, use a nearest neighbour heuristic to generate a route going through all mailboxes originally located on street edges and one copy (chosen arbitrarily) of every box located at an intersection. The two ends of the route usually correspond to two points located near the "extremities" of the cluster. They are often specified by the user. Improve this route by means of the 2-opt procedure used in Bodin and Berman (1979).

Step 5 **Updated clustering.** Check whether the total distance travelled can be reduced by performing marginal modifications to the clusters, as in step 3. This procedure is repeated as long as distance reductions can be achieved. At the end of this step, cluster C_p contains n'_p stops. This clustering method tends to produce solutions having a small inter-stop distance, but may involve a large amount of line-haul travel and does not always minimize total distance when

there exist clusters which are not adjacent to the depot. Further improvements could therefore be possible, along the lines suggested in some recent studies (see, for example, Haimovich and Rinnoy Kan (1985), Daganzo and Newell (1986) and Langevin and Soumis (1988)).

Step 6 **Route reoptimization.** In this step, duplicated mailboxes are replaced by their copy if this saves distance. Formally, this is done as follows. Initially, each route can be described by a sequence of n'_p stops of the form $(i_1,\ldots,i_{n'_p})$. Now consider all subsequences of the form $(i_s,i_{s+1},\ldots,i_t)$ where i_s, $i_t \in I$, $i_{s+1},\ldots,i_{t-1} \in J_1 \cup J_2$. Let i'_j denote the copy of i_j (if $i_j \in J_1$, then $i'_j \in J_2$, and conversely). A complete search is made to determine whether distance reductions could be achieved by replacing any i_j by i'_j for $s+1 \leq j \leq t-1$. Since all possibilities are examined, this involves a total of 2^{t-s-1} evaluations. Fortunately, in practice, the value of t-s-1 is never very large (it rarely exceeds 5).

4. GENERALIZED TRAVELLING SALESMAN PROBLEM FORMULATION

One disadvantage of the above procedure is that once routes have been generated after step 5, they can only be improved marginally by replacing duplicated mailboxes by their copy. The relative order of stops, however, stays the same. CPC analysts felt that an improvement could be achieved if it was possible to integrate the route generation procedure and the selection of the "best copy" of all mailboxes originally located at street intersections. We show that, provided the cluster cardinalities are kept relatively small, this problem can be solved to optimality with reasonable computing effort. Consider a cluster with its original mailboxes and

(i) consider a street side (between two intersections) where one or more mailboxes are located; to this street side corresponds an arc (i,j); define a set $S_a = \{a\}$ for the a^{th} such arc;

(ii) for each mailbox ℓ located at an intersection, define a set $T_\ell = \{\ell, \ell'\}$ where ℓ and ℓ' are the two copies of the mailbox.

Also define $S = U_a S_a$ and $T = U_\ell T_\ell$. At the end of this operation, we have created n' sets and $r = |S| + |T|$ nodes which are labelled consecutively. Distances between the nodes of S U T are defined as follows: if a=(i,j) and a'=(i',j') both belong to S, then $c_{a,a'}$ is the shortest distance from j to i'; if $a=(i,j) \in S$ and $k \in T$, then $c_{a,k}$ is the shortest distance from j to k and $c_{k,a}$ is the shortest distance from a to i. Our aim is to determine the shortest route starting and ending at two given nodes, and passing through one node of each set. Introduce a dummy node $i_0 = r+1$ with the corresponding set $V = \{i_0\}$. Define a zero distance between i_0 and the two end nodes and an infinite distance between i_0 and all other nodes. The problem is then a **generalized travelling salesman problem** (GTSP) through $n = n'+1$ sets of nodes.

The GTSP constitutes an extension of the travelling salesman problem (TSP) (see Dantzig et al. (1954) for example); in the TSP, all sets of nodes have a cardinality of one. The GTSP has been studied by Laporte and Nobert (1983) for symmetrical distance matrices and more recently by Laporte et al. (1987) for asymmetrical distance matrices, as is the case in this application.

The formulation of the (asymmetrical) GTSP particularized to our problem is as follows. Let N = S U T U V be the set of all nodes and $C = (c_{i,j})$ the matrix of shortest distances between pairs of nodes of N.

Let $x_{i,j}$ be 0-1 variables indicating whether arc (i,j) belongs to the optimal solution ($x_{i,j}= 1$) or not ($x_{i,j}= 0$), $i,j \in N$ and $i \neq j$. Let $x_{i,i}$ be other 0-1 variables indicating whether node i appears in the optimal solution ($x_{i,i}= 0$) or not ($x_{i,i}= 1$). Variables $x_{i,i}$ are only defined for $i \in T$ and must be interpreted as zero otherwise. The problem is then to

$$(P) \qquad \text{minimize} \quad z = \sum_{\substack{i,j \in N \\ i \neq j}} c_{i,j}\, x_{i,j} \qquad (1)$$

subject to

$$\sum_{i \in N} x_{i,k} = \sum_{j \in N} x_{k,j} = 1 \qquad (k \in N) \qquad (2)$$

$$\sum_{i \in T_\ell} x_{i,i} = 1 \qquad (\text{all } T_\ell) \qquad (3)$$

$$\sum_{\substack{i,j \in M \\ i \neq j}} x_{i,j} \leq |M| - 1 \qquad (M \subset N;\ M \neq T_\ell,\ \text{all } \ell) \qquad (4)$$

$$x_{i,j} = 0 \text{ or } 1 \qquad (i,j \in N) \qquad (5)$$

In this formulation, constraints (2) ensure that every node of S U V is entered and left once and exactly once. When applied to nodes k of T, these constraints ensure that k is entered and left once if and only if $x_{k,k}= 0$, i.e. if k appears in the optimal solution. Constraints (3) specify that in every T_ℓ, one node out of two is visited. Constraints (4) are the classical subtour elimination constraints originally developed by Dantzig et al. (1954) for the TSP. They specify that in any subset of nodes of N, the number of arcs must be strictly less than the number of nodes, thus preventing the formation of tours involving less than |N|

nodes. It is not necessary to impose such constraints for $M = T_\ell$ since the two nodes of T_ℓ can never enter the solution simultaneously. Finally, constraints (5) specify the binary conditions on the variables.

5. THE GTSP ALGORITHM

The GTSP algorithm used for the solution of (P) constitutes the seventh and final step of the clustering and routing procedure. It is applied to each cluster C_p independently and is used as a post-optimizer (after step 6), once clusters guaranteeing reasonable route lengths have been defined. However, since the GTSP algorithm can only solve relatively small size problems in a reasonable time, we have opted to apply it to clusters involving up to about 30 nodes, corresponding roughly to 20 original mailbox locations, 10 of which are duplicated. With this in mind, two approaches are possible for the definition of the clusters in step 2 of the algorithm: (i) ensure that $|C_p| \leq 20$ for all p and later generate complete vehicle routes by merging two or three of the shorter routes; (ii) define the clusters in such a way that they include as many mailboxes as normally constitute a vehicle route; since these routes will often involve as many as 45 to 50 different stops, they will have to be broken down into two or three segments before applying the GTSP algorithm. We have opted for the second approach after having observed that (long) vehicle routes very often have "natural breaking points", corresponding to relatively long driving stretches between small clusters of stops.

The GTSP algorithm (fully described in Laporte et al. (1987)) was constructed by extending and otherwise modifying the Carpaneto and Toth (1980) branch and bound algorithm for asymmetrical TSP's. At every node of the search tree, this algorithm solves an assignment problem (AP) in which some variables are forced to take the value 0 or 1. At a given node h, a check is made to determine whether the current solution is legal or not. A

solution may be illegal for one or both of the following reasons: (i) at least one T_ℓ is not visited; (ii) illegal subtours exist. Illegal solutions are eliminated by generating subproblems in which the appropriate variables are fixed at 0 or at 1 (we used here the Carpaneto and Toth partitioning rule). The remainder of the algorithm follows the usual branch and bound methodology.

6. COMPUTATIONAL RESULTS

The algorithm was coded in FORTRAN and tested over 10 clusters in two postal regions of the Montreal area. Computational results are reported in table 1. They compare the solution value obtained by applying the GTSP algorithm as a post-optimizer (column labelled (7)) to those obtained after steps (5) and (6) of the current CPC algorithm. The results indicate that the GTSP step can be applied provided the number of nodes in a cluster is not too large. Memory capacity (350K) was exceeded in two problems (2 and 7) involving 27 and 26 nodes respectively. These are the two problems where the largest number of mailboxes were duplicated: in such problems, the number of times constraints (3) were violated in the course of the branch and bound algorithm tended to be high. This generated an excessive number of subproblems and the algorithm could not be completed. Otherwise, the results are very positive and computation times are in most cases reasonably low. On the ten test problems, applying the GTSP algorithm resulted in distance reductions ranging from 0% to 14.5%. The discrepancies appearing in column 7 of table 1 (percentage improvement) were examined, but could not be related in any obvious way to problem characteristics. The variability in improvement percentages is more a reflection of the degree of suboptimality of the CPC algorithm than of the quality of the GTSP algorithm which performs consistently well and always provides the optimum, when run to completion. Since mailbox collections are carried out several

times a day in a large number of Canadian cities, the reductions observed could translate into large savings in the long run.

Finally, some of the itineraries produced after steps 6 and 7 of the algorithm were examined and compared. An example is provided in figures 2 and 3. These diagrams show that the collection routes can become at times quite intricate. This is due to the presence of the large number of one-way streets and left-turn prohibitions in a city such as Montreal. It is difficult, by visual inspection of figure 2, to see obvious improvements to the collection routes; only a sophisticated algorithmic tool can provide and guarantee the best possible improvement when one exists. The whole procedure described in this paper was developed by assuming fixed mailbox locations. By examining the two routes illustrated in figure 3, one can see several instances where minor changes in most mailbox locations (across the street, for example) would yield large distance savings. It is relatively straightforward to include in the GTSP algorithm the possibility of determining optimal mailbox locations (on either side of the street, for example – see Labbé and Laporte (1986)), together with the associated optimal routing. However, as mentioned in section 2, mailbox relocations are seldom possible.

7. CONCLUSION

We have described in this paper a revised algorithm for the mailbox collection problem. Limited computational tests indicate that its use could result in substantial savings at CPC and an implementation contract has now been awarded. The algorithm is also well-suited to integrate mailbox location to route design. A further related problem now investigated by our team is the optimal location of relay-(or transfer) boxes used by letter-carriers. Since mailboxes and relay-boxes are usually visited in one trip by the same truck, the two location-routing problems should ideally be solved jointly.

Table 1: Summary of Computational Results

Cluster	Number of mailboxes	Number of nodes after duplications	Solution value (in meters) after step			Percentage improvement [(6)-(7)]÷(6)	CPU time (seconds) (*)
			5	6 last step of CPC algorithm	7 GTSP step		
1	14	22	15923	14397	13931	3.2%	94
2	15	27	15713	15449	15449	0.0%	(**)
3	13	22	14293	14293	12225	14.5%	252
4	10	18	9219	7942	7671	3.4%	31
5	19	26	8643	8643	8643	0.0%	18
6	12	20	7735	7290	7290	0.0%	19
7	15	26	12204	11642	11237	3.4%	(**)
8	11	17	10297	10140	9644	4.9%	23
9	13	22	9837	9285	8000	13.8%	31
10	13	24	11948	10870	9808	9.7%	80

(*) Université de Montréal CYBER 855 computer. (All CPU times are CYBER 173 equivalent.)
(**) Memory requirements exceeded. Optimality not proven.

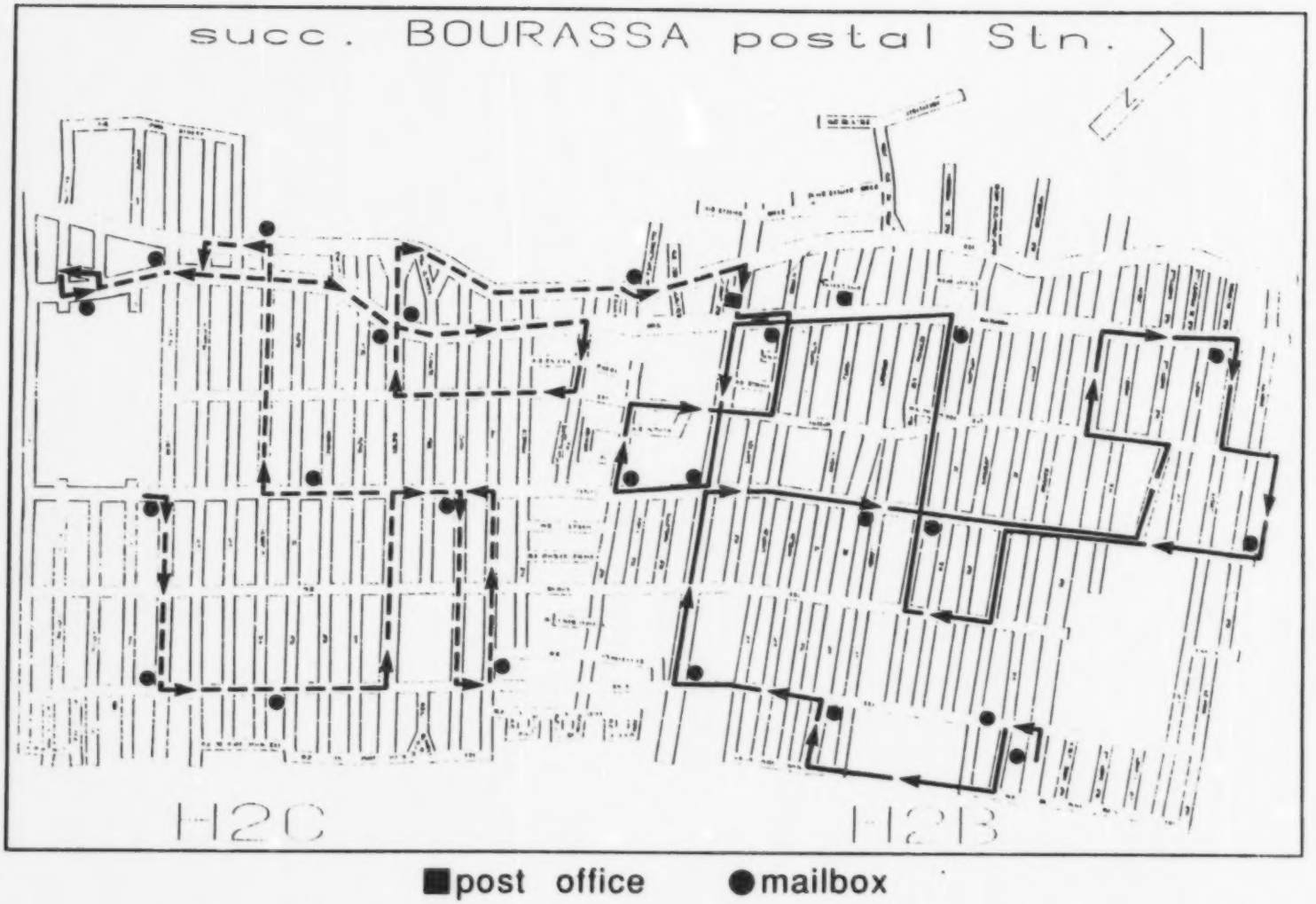


Figure 2: Best itineraries determined by the CPC algorithm (after step 6) in clusters 9 (dotted line) and 10 (full line)

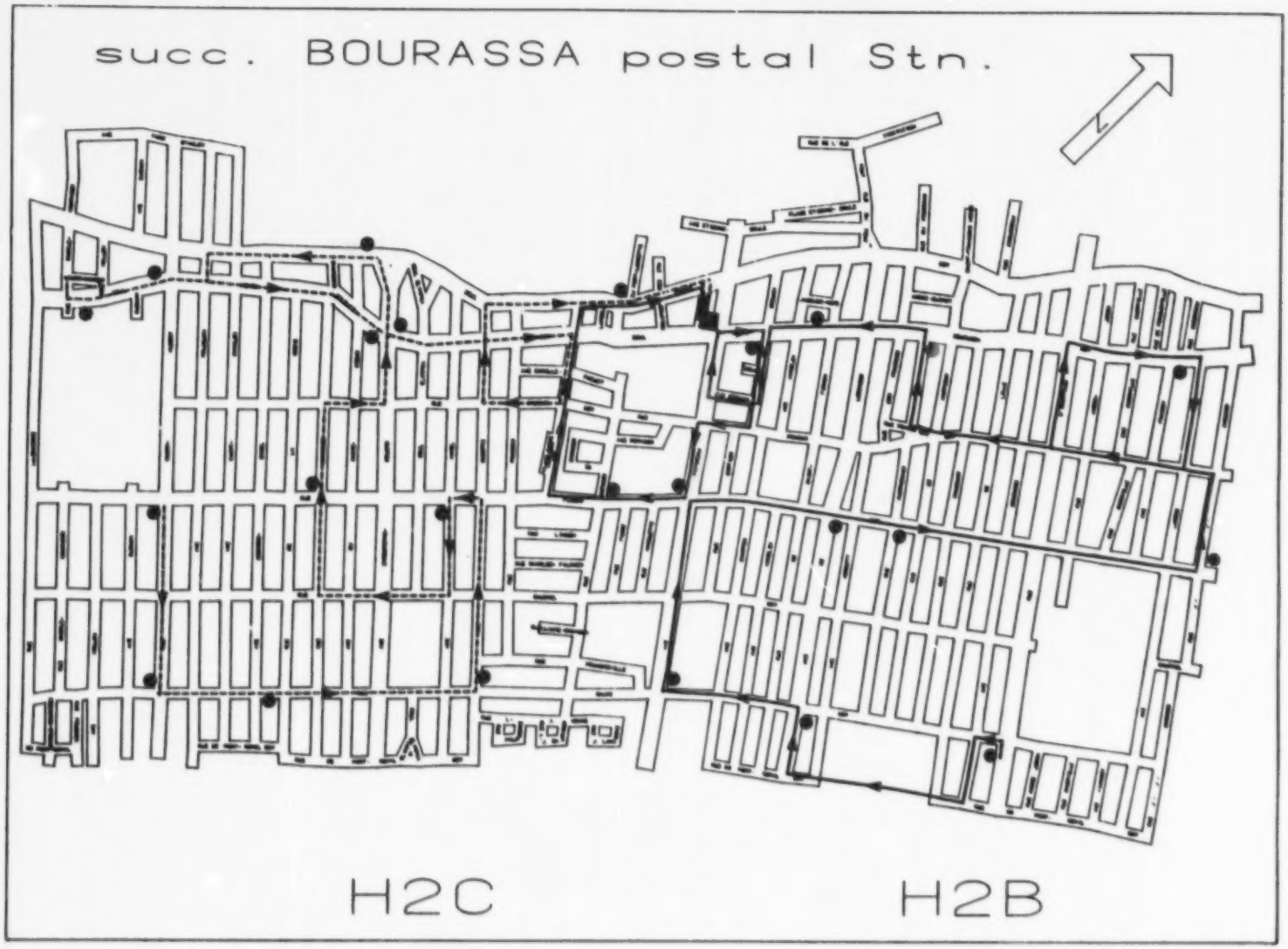


Figure 3: Itineraries determined by applying the GTSP algorithm (step 7) as a post-optimizer in clusters 9 (dotted line) and 10 (full line)

REFERENCES

BODIN, L.D., BERMAN, L. (1979) "Routing and Scheduling of School Buses by Computer", Transportation Science 13, 113-129.

BODIN, L.D., GOLDEN, B.L., ASSAD, A.A., and BALL, M.O. (1983) "Routing and Scheduling of Vehicles and Crews: The State of the Art", Computers and Operations Research 10, 69-211.

BODIN, L.D., LEVY, L.S. (1987) "The Walking Line of Travel Problem: An Application of Arc Routing and Partitioning", working paper #87-024, College of Business and Management, University of Maryland at College Park, MD.

BOURJOLLY, J.-M., LAPORTE, G., ROUSSEAU, J.-M. (1981) "Découpage électoral automatisé : application à l'Ile de Montréal", INFOR 19, 113-124.

CARPANETO, G., TOTH, P. (1980) "Some New Branching and Bounding Criteria for the Asymmetrical Travelling Salesman Problem", Management Science 26, 736-743.

DAGANZO, C.P., NEWELL, G.F. (1986) "Configuration of Physical Distribution Networks", Networks 16, 113-132.

DANTZIG, G.B., FULKERSON, D.R., JOHNSON, S.M. (1954) "Solution of a Large Scale Traveling Salesman Problem", Operations Research 2, 393-410.

HAIMOVICH, M., RINNOOY KAN, A.H.G. (1985) "Bounds and Heuristics for Capacitated Routing Problems", Mathematics of Operations Research 10, 527-542.

LABBÉ, M., LAPORTE, G. (1986) "Maximizing User Convenience and Postal Service Efficiency in Post Box Location", Belgian Journal of Operations Research, Statistics and Computer Science 26, 21-36.

LANGEVIN, A., SOUMIS, F. (1988) "Design of Multiple-Vehicle Delivery Tours Satisfying Time Constraints", Transportation Research B (forthcoming).

LAPORTE, G., NOBERT, Y. (1987) "Exact Algorithms for the Vehicle Routing Problem", in S. Martello, G. Laporte, M. Minoux and C. Riberio editors, Surveys in Combinatorial Optimization, Annals of Discrete Mathematics 31, North-Holland, Amsterdam, 147-184.

LAPORTE, G., MERCURE, H., NOBERT, Y. (1987) "Generalized Travelling Salesman through n Sets of Nodes: The Asymmetrical Case", Discrete Applied Mathematics 18, 185-197.

LAPORTE, G., NOBERT, Y. (1983) "Generalized Travelling Salesman through n Sets of Nodes: An Integer Programming Approach", INFOR 21, 61-75.

LEVY, L.S. (1987) The Walking Line of Travel Problem: An Application of Arc Routing and Partitioning, Ph.D. Dissertation, College of Business and Management, University of Maryland at College Park, MD.

LEVY, L.S., BODIN, L.D. (1988a) "The Arc Oriented Location Routing Problem", submitted for publication.

LEVY, L.S., BODIN, L.D. (1988b) "Scheduling the Postal Carriers for the United States Postal Service: An Application of Arc Partitioning and Routing", in Vehicle Routing: Methods and Studies, B.L. Golden and A.A. Assad, eds., North-Holland, Amsterdam, 359-394.

ROY, S., ROUSSEAU, J.-M. (1988) "The Capacitated Canadian Postman Problem", submitted for publication.